

Case Study - Jungles

BATTLEFIELD VIETNAM

Riccard Linde

Art Director

Digital Illusions Canada Inc



Today's Menu

- Me - Introduction
- Art Direction – Battlefield Vietnam.
- Jungle – millions of trees.
- Grass – under vegetation.
- Bringing it all together in the engine.
- Troubleshooting Battlefield Vietnam.
- Questions



Riccard Linde

- From Sweden - 27years old
- 12 years in CG industry.
- 3 years in the game industry.
- 4 shipped titles.
- Lead Artist for BF1942 expansion packs Road To Rome and Secret Weapons of WWII. Art Director for BF Vietnam.
- Thanks! GDC. Team DICE Sweden & Team DICE Canada.



BATTLEFIELD
VIETNAM

Shipped – March 2004



Art Direction

- **Real world environments – With ‘CG’ textures. Drawn from real photos or references.**
- **Represent Hollywood movies.**
- **Vietnam environment was NOT the main source for references.**

BATTLEFIELD
VIETNAM



Vietnam through Hollywood



BATTLEFIELD
VIETNAM



Vietnam's new Gfx engine!

- **Per pixel Normal mapping (world space, tangent) and specular calculation mapping.**
- **DirectX9. Reflections, fresnel shaders.**
- **Multiple environment lights.**
- **Tree system for more believable jungles.**



Overgrowth

Goal – To be able to create and tweak a tropical jungle.



To consider!

1. Performance

- Lots of information added throughout the whole level.

2. Art assets

- Look good and blend into the world.

3. Game Play

- To many trees – Vehicles are supposed to be able to drive here!





Art Assets

- 1. HIGH resolution trees**
 - Main visible geometry.
- 2. LOW resolution trees**
 - Less detail = more trees.
- 3. Patch of cluster trees**
 - Treated as one object.



HIGH resolution trees



- ~7m/diameter Clusters.
- 400-2000 polys.
- 1 Shader/Textures.
- No Alpha blend.
- 2 side modeled leafs.
- Fade HIGH <> LOW.



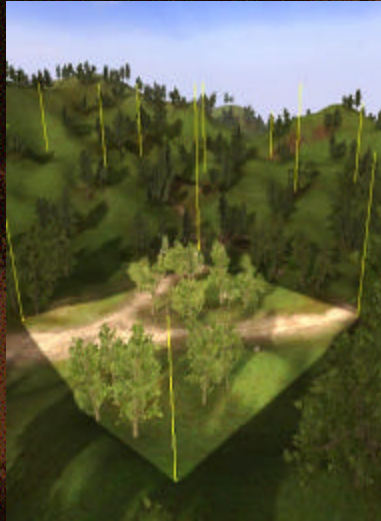
LOW resolution trees



- Representation of HIGH resolution mesh.
- <100 Vertexes.
- 1 Shader/Texture.
- No Alpha blend.
- Fade HIGH <> LOW.



Patch of Cluster trees



- Engine merges from LOW res. One object in divided patches.
- 2048 vertex limit /patch
- 1 Shader/Texture
- No Alpha blend



How big is a patch?

- View distance of trees decide size of the patch.
- Larger patch = greater spread of the trees.





Working with the tools!

- **Automatic placement.**
 - Script and material bitmap.
- **Manual placement.**
 - Do it your self.



Automatic Placement



- **Scripts to specify material ID's – angle, spread, scale, spacing and the type of trees are stated.**
- **Bitmap (4bits Raw file) to decide where placement of the different material are in the world.**



Script for define placement!

```
<WRAPPER_TREE
  VERS = "1.1">
  <overGrowth
    materialMapSideSize = "512"
    viewDistance = "550"
    importSceneObjects = "true">
    <materials>
      <dryGrass>
        <types>
          <c02f_trees_m2
            geometryName = "c02f_trees_m2"
            minRadiusDistToEquals = ".5"
            minRadiusDistToOthers = ".25"
            normalScale = ".3"
            probability = ".7"
            scale = "CRDUniform/.8/1.2/false">
          </c02f_trees_m2>
        </types>
      </dryGrass>
    </materials>
  </overGrowth>
</WRAPPER_TREE>
```



4bit 512*512 Bitmap = where and what?!





Manual placement in editor



- Place and remove where needed for game play.
- Control Rotation, position and scale.
- Merged together with the engines auto generated trees.
- Placed trees has a higher priority, removes the auto generated trees if the Vertex limit is full.





Undergrowth

**Goal – Add short vegetation,
make the world look less flat.**



Art Direction & System

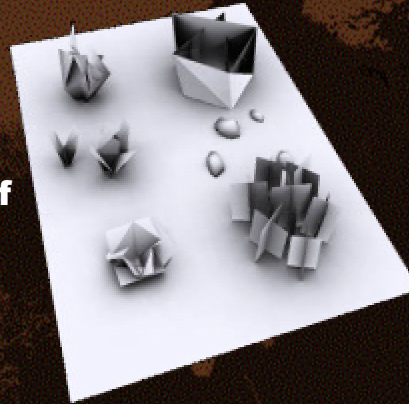
- **The grass should never POP in at any distance, blend seamless.**
- **Had to inherit shadow color on ground from trees and objects.**
- **The same tech as the overgrowth. Script and 4bit material bitmap file.**





Low Resolution Meshes

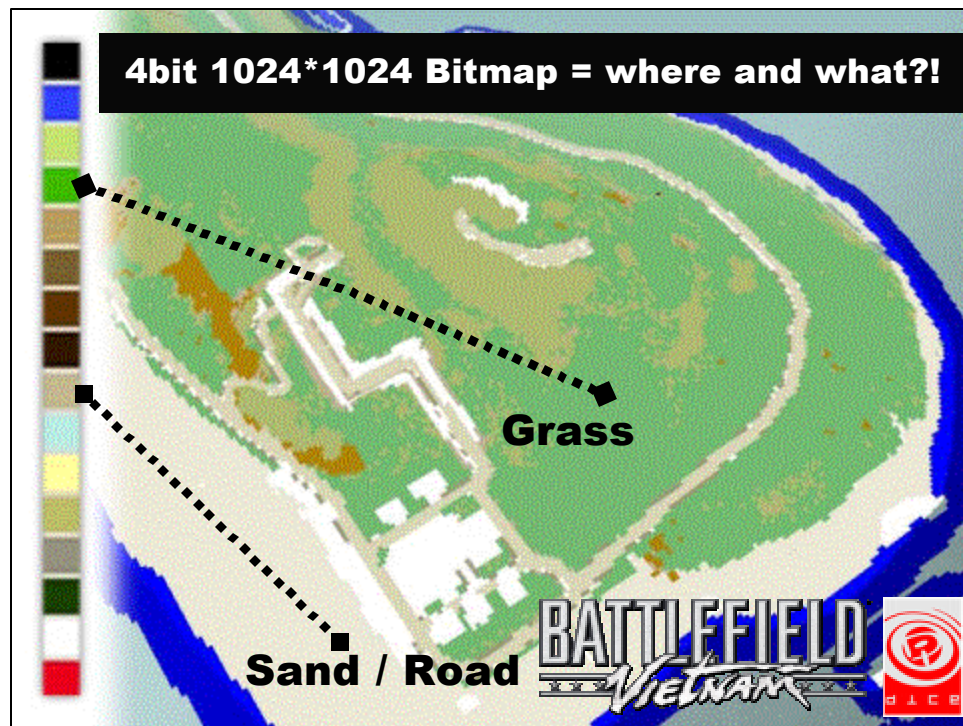
- <100 polygons
- 1 Shader/Texture
- Alpha blending
- Bitmap (4bits) decides what kind of grass goes where.
- View distance determents density.
- Mixes diffuse color with ground map to blend in.



Script for define placement!

```
<WRAPPER_TREE
  VERS = "1.1">
  <underGrowth
    materialMapSideSize = "1024"
    viewDistance = "60"
    <materials>
      </Grass>
      <types>
        <F_undergrowthG_m1
          geometryName = "F_undergrowthG_m1"
          minRadiusDistToEquals = "4"
          minRadiusDistToOthers = "3"
          normalScale = "1"
          probability = "0.1"
          scale = "CRDUniform/1/2/false">
        </F_undergrowthG_m1>
      </types>
    </Grass>
  </materials>
</underGrowth>
</WRAPPER_TREE>
```







JUNGLES

Inside the Engine



Trouble shooting – Next version!

- **View distance changes or movement of trees = redo render on ground shadows.**
- **Miss matches between low and high resolution meshes – Carefully modeling from m2 as a base.**
- **2 shaders would have been a better choice – on for alphas, and on for trunk, solid geometries.**
- **The ability to let them know if they stand in shadows or not on the ground to make them blend in more.**



Trouble shooting – Gameplay?

- **Over growth density, stopped vehicles.**
- **Trees blocked the player view. Where are the players?. Where do I get shot from?**
- **Trees camouflaged the flags, players were confused when flag could not be found amongst trees in the bases.**
- **Undergrowth – Hidden close by, visible from the next hill.**



How else did we use it?

- **Place other ‘one draw call’ objects inside houses and in the world with the help of the overgrowth system.**



Variations of placed objects!



?Questions?

